

High quality voice conversion using prosodic and high-resolution spectral features

Hy Quy Nguyen · Siu Wa Lee · Xiaohai Tian ·
Minghui Dong · Eng Siong Chng

The final publication is available at www.springerlink.com.

Submitted to Multimedia Tools and Applications: 7 June 2015 / Accepted: 20 October 2015 / Published as 'Online First' on SpringerLink: 22 Nov 2015

Abstract Voice conversion methods have advanced rapidly over the last decade. Studies have shown that speaker characteristics are captured by spectral feature as well as various prosodic features. Most existing conversion methods focus on the spectral feature as it directly represents the timbre characteristics, while some conversion methods have focused only on the prosodic feature represented by the fundamental frequency. In this paper, a comprehensive framework using deep neural networks to convert both timbre and prosodic features is proposed. The timbre feature is represented by a high-resolution spectral feature. The prosodic features include F0, intensity and duration. It is well known that DNN is useful as a tool to model high-dimensional features. In this work, we show that DNN initialized by our proposed autoencoder pretraining yields good quality DNN conversion models. This pretraining is tailor-made for voice conversion and leverages on autoencoder to capture the generic spectral shape of source speech. Additionally, our framework uses segmental DNN models to capture the evolution of the prosodic features over time. To reconstruct the converted speech, the spectral feature produced by the DNN model is combined with the three prosodic features produced by the DNN segmental models. Our experimental results show that the application of both prosodic and high-resolution spectral features leads to quality converted speech as measured by objective evaluation and subjective listening tests.

Keywords voice conversion · deep neural network (DNN) · spectral transformation · fundamental frequency (F0) · duration modeling · pretraining

Hy Quy Nguyen · Xiaohai Tian · Eng Siong Chng
School of Computer Engineering, Nanyang Technological University (NTU), Singapore
Joint NTU-UBC Research Centre of Excellence in Active Living for the Elderly, NTU, Singapore
E-mail: ng0002hy@e.ntu.edu.sg, xhtian@ntu.edu.sg, aseschn@ntu.edu.sg

Siu Wa Lee · Minghui Dong
Human Language Technology Department, Institute for Infocomm Research, A*STAR, Singapore
E-mail: swylee@i2r.a-star.edu.sg, mhdong@i2r.a-star.edu.sg

1 Introduction

Voice conversion modifies the recorded speech of a source speaker toward a given target speaker. The resultant speech has to sound like the target speaker with the language content unchanged. To achieve this, conversion functions are applied to source speech features, such as timbre and prosodic features like fundamental frequency (F0), and so on. Over the years, most of the research focuses in the area of spectral mapping, i.e. conversion of the timbre characteristics of the source to those of the target speaker. Spectral mapping on low-dimensional spectral envelope representation has become mature recently [39] and started moving to detailed spectral representation [45], [4]. On the other hand, psychoacoustics, linguistics, text-to-speech (TTS) and speaker recognition studies have showed that prosodic features such as F0, energy contour and duration at various levels of speech also carry information on speaker characteristics [2], [6], [8], [1], [31]. For example, many politicians and celebrities are well known by their timbres, intonation, speaking rates and choices of words, etc. Unlike spectral mapping, there are not many established works on prosodic features yet. In the following, a comprehensive voice conversion approach utilizing various aforementioned features will be introduced. It is ‘comprehensive’ in the sense that various prosodic features are adopted acoustically, besides spectrum. Based on the characteristics, each individual feature is modeled and converted in different levels respectively. To facilitate non-linear, high-resolution transformation, our approach is built on deep neural networks (DNNs).

Conventional spectral mapping in voice conversion falls into one of the two categories: Gaussian mixture model (GMM) [34] and frequency warping (FW). GMM-based conversion [34], [12], [11] are statistical approaches, where the joint likelihood between the source and target spectra is maximized and the conversion functions are estimated accordingly. As low-dimensional features and statistical conversion are used, these methods are computationally-inexpensive and robust. However, spectral details are usually lost in low-dimensional representations, and the statistical averaging during training often leads to over-smoothed speech outputs. To reduce over-smoothing, techniques such as global variance [39], [15] were introduced.

FW is another technique to tackle over-smoothing [35], [10], [37], [9], [38]. By limiting the conversion to only warping the frequency axis of a high-dimensional source spectrum towards the target spectrum, spectral details are preserved. Nevertheless, the estimation of the exact warping functions is not straightforward [37]. The newly-emerging exemplar-based approach [36], [44], [45] operates on high-dimensional features too. As a few basis spectral features are used and combined to form the output spectrum, the over-smoothing issue in GMM-based approach therefore no longer exists.

Recently, DNN-based techniques have been well presented in speech community [50], [23]. Voice conversion using DNN models generates non-linear mapping between source and target features, and there is little restriction in the feature dimensions to be modeled. A very early work on DNN-based spectral conversion focusing formant transformation was presented in [28]. Some other pioneering works in spectral conversion include [7] and [26]. Other works employed Restricted Boltzmann Machines (RBM) to estimate joint distribution [5], [43] or to estimate high level fea-

tures [27], [3]. A recent work on spectral conversion [46] used DNN to perform transformation directly on high-dimensional spectral features, delivering accurate conversion and decent speech quality. Voice conversion using multiple speaker input has also been investigated in [24]. Our proposed voice conversion on various prosodic and spectral features is built by leveraging on the above merits of DNNs.

Speaker characteristics are carried by various speech features. Besides spectrum, comprehensive voice conversion requires other features for transformation, for example, F0. Nevertheless, unlike spectrum, there are not many established works on F0 conversion. Some methods include transformation using vector quantization [13], partial least square regression [30] and DNN mapping [47]. Among these, [30] and [47] have focused on wavelet domain instead of on the frequency domain directly. The most popular method is the mean-variance global transformation, adjusting the average level and the range of source F0 values [34]. This method retains the general shape of the source F0 trajectory and ignores the detailed difference between the two F0 contours. It is a frame-level operation, while human manipulates F0 in a segmental manner with a scope which may be up to phone, syllable, word, phrase or sentence level. F0 modeling and generation becomes challenging in voice conversion, as the amount of training data is usually sparse with tens to a few hundreds of utterances only.

Other prosody characteristics such as duration and intensity have been investigated in expressive speech synthesis and voice conversion. Representative works include duration modification based on HMM model [42], mean-variance transformation for phone or utterance duration [33], state duration modification using decision tree [49] and duration embedded GMM-based conversion [52]. Most of these methods use phone boundaries or phone identities from text labels. In [52], consecutive spectral frames are used for duration modeling. Studies have also suggested some ways to perform intensity conversion based on F0 to improve naturalness [32].

In this work, we explore the feasibility of building a comprehensive voice conversion framework. Specifically, our contributions include the following: (1) Various features with spectrum information and those with prosody information such as F0, intensity and phone duration are modified under a comprehensive DNN framework. Our spectral conversion operates directly on high-resolution full spectra and the frame-level mapping is estimated; (2) We propose a new pretraining scheme based on autoencoder, to further improve the training of the DNN model under the condition of limited training data in common voice conversion applications; (3) On one hand, unlike spectrum, prosodic features such as F0, intensity and duration depend very much on the semantic meaning and speaking style, which in turn depend on suprasegmental information [25]. On the other hand, most voice conversion systems work on limited number of training utterances, which makes it hard to model suprasegmental context. In this work, we move from the conventional frame-level modeling and propose a segment-level modeling scheme, aiming to balance between the context length and the amount of training data. Particularly, the input and output for F0 and intensity are based on every contiguous voiced segment, rather than a feature frame or short moving window; (4) Our F0 conversion is aimed at the overall F0 trajectory, instead of single F0 values. Direct modeling of F0 difference between adjacent frames is used to emphasize the F0 movement over time and consequently similar F0 trajectories

can be compactly modeled. This facilitates the learning of DNN model parameters, which is crucial for the common scenario of limited voiced segments in training data of voice conversion. Finally, similar to our F0 modeling, duration ratio is modeled segmentally with input patterns constructed with sampled spectral frames.

The above proposed framework has been examined in details. The conversion performance delivered by various features was verified. In particular, the effectiveness of our autoencoder pretraining is assured and an outstanding performance in objective measurement was achieved. Our comprehensive conversion framework was shown to provide the highest rating on speaker similarity.

In the following sections, we will briefly review GMM-based and DNN-based voice conversion systems, which are two state-of-the-art and are also our baselines for evaluation.

2 GMM-based Voice Conversion

Joint Density GMM (JD-GMM) [39], [16] is one of the most successful methods in GMM-based voice conversion. In this method, GMM is estimated to model the joint distribution of source speech \mathbf{X} and target speech \mathbf{Y} . During training, expectation-maximization (EM) method is used to maximize the joint probability as below:

$$P(\mathbf{X}, \mathbf{Y}) = P(\mathbf{Z}) = \sum_{k=1}^K w_k \mathcal{N}(\mathbf{z} | \boldsymbol{\mu}_k^{(z)}, \boldsymbol{\Sigma}_k^{(z)}) \quad (1)$$

$$\mathbf{z} = \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix}, \quad \boldsymbol{\mu}_k^{(z)} = \begin{bmatrix} \boldsymbol{\mu}_k^{(x)} \\ \boldsymbol{\mu}_k^{(y)} \end{bmatrix}, \quad \boldsymbol{\Sigma}_k^{(z)} = \begin{bmatrix} \boldsymbol{\Sigma}_k^{(xx)} & \boldsymbol{\Sigma}_k^{(xy)} \\ \boldsymbol{\Sigma}_k^{(yz)} & \boldsymbol{\Sigma}_k^{(yy)} \end{bmatrix} \quad (2)$$

where \mathbf{Z} is the joint paired feature vector sequence, K is the number of Gaussian components, w_k is the weight of the k -th component and $\mathcal{N}(\mathbf{z} | \boldsymbol{\mu}_k^{(z)}, \boldsymbol{\Sigma}_k^{(z)})$ is the Gaussian distribution with mean $\boldsymbol{\mu}_k^{(z)}$ and covariance matrix $\boldsymbol{\Sigma}_k^{(z)}$ of the k -th component.

During conversion, minimum mean square error is employed to estimate the target feature vector $\hat{\mathbf{y}}$ from each input source feature vector \mathbf{x} :

$$\hat{\mathbf{y}} = F(\mathbf{x}) = \sum_{k=1}^K p_k(\mathbf{x}) [\boldsymbol{\mu}_k^{(y)} + \boldsymbol{\Sigma}_k^{(yx)} (\boldsymbol{\Sigma}_k^{(xx)})^{-1} (\mathbf{x} - \boldsymbol{\mu}_k^{(x)})] \quad (3)$$

$$p_k(\mathbf{x}) = \frac{w_k \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k^{(x)}, \boldsymbol{\Sigma}_k^{(xx)})}{\sum_{k=1}^K w_k \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k^{(x)}, \boldsymbol{\Sigma}_k^{(xx)})} \quad (4)$$

where $p_k(\mathbf{x})$ is the posterior probability of the source vector \mathbf{x} generated from the k -th Gaussian component.

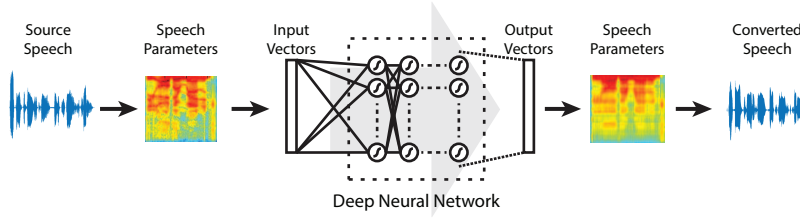


Fig. 1 DNN-based Voice Conversion

3 DNN-based Voice Conversion

In this approach, a DNN is used to model the transformation from source to target speech features as demonstrated in Fig. 1. Most of the state-of-the-art DNN-based voice conversion methods are on spectral transformation. In those methods, spectral parameters of the source speech are extracted and put into an input vector for conversion. This input vector is then passed to a DNN-based conversion network, where the network performance is governed by the network architecture, node weights and biases and the associated activation functions. By feeding forward the input vector through the network, an output vector is generated which represents the converted speech parameters. These output parameters are finally used to construct the converted waveform. Conversion on other features, such as F0, can also be done similarly.

However, the appropriate set of DNN model parameters that transforms source to target feature vectors is unknown at the beginning. A training phase is required. Before model training, random initialization on the model parameters is applicable or a pretraining step using layer-wise restricted Boltzmann machine (RBM) pretraining [4] or discriminative pretraining [51] can be performed. During model training, the error measurement of each hidden node is calculated using back-propagation algorithm [29], and the DNN model parameters are updated accordingly to optimize a training criterion. A common training criterion in voice conversion is the sum of squared error between the target output vector and the model output. At the conversion phase, input vectors are propagated forward through the model with these estimated parameters to produce the corresponding output vectors.

One of the pioneering work in voice conversion using DNN is [7] in which a low-dimensional spectral representation, mel-cepstral coefficients (MCEP), is used directly as input and output vectors for a feed forward DNN. Another work [26] employs Deep Belief Nets (DBNs) to extract latent features from source and target cepstrum coefficients, and uses a neural network with one hidden layer to perform conversion between latent features. High-dimensional representation of spectrum has also been used in a more recent work [46] for spectral mapping, together with dynamic features and parameter generation algorithm [40]. Moreover, DNN model has also been used to generate the F0 of target speaker with input spectral parameters from the source speaker [47].

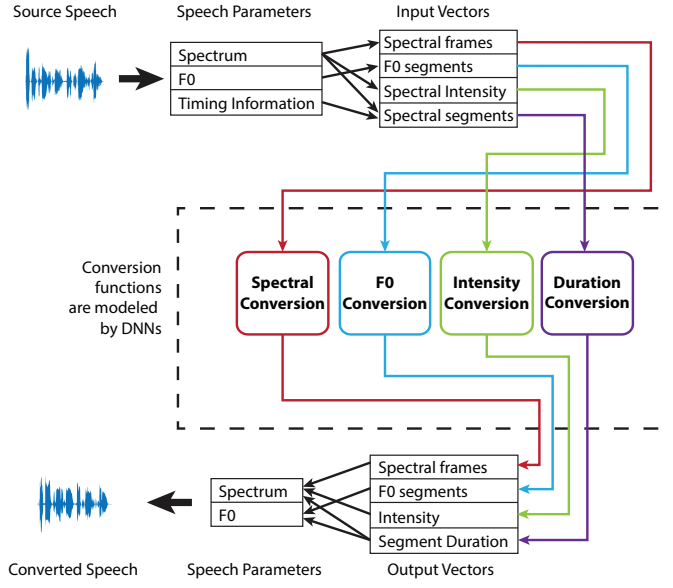


Fig. 2 Comprehensive Voice Conversion Framework

4 Proposed Conversion Framework

We now propose a comprehensive framework where spectrum, F0 contour, intensity trajectory and phone duration are all modeled using feed-forward DNNs. Due to distinct natures of various features (frame-level and segment-level), we used separate DNN model for each input feature. The framework is demonstrated in Fig. 2.

We extracted speech features from the input source speech and create appropriate input vectors for each conversion system. Each system is modeled by a DNN, of which parameters are estimated from the set of training data with the back-propagation algorithm and mean square error criteria. The converted outputs from each DNN model of individual features will be finally combined to reconstruct the output converted speech waveform. Detailed specifications of each model will be introduced below.

4.1 Spectrum Conversion

We use a feed-forward DNN to model the relationship between the source and target spectra. The DNN input is a source spectrum frame with dynamic features as context, while the DNN output is the corresponding target spectrum frame. In the current work, for simplicity, we do not employ parameter generation algorithm [40] with delta and delta-delta at output layer.

Our spectrum conversion process is partly based on [46]. To improve the conversion performance further from the above DNN-based conversion framework, we have made two modifications described below.

4.1.1 Two-stage Alignment

During data preparation, a two-stage alignment process is performed [22]: Each training utterance are cut into phone segments using a phone recognizer in the first stage and then corresponding individual phone segments of source and target speaker are aligned using dynamic time warping (DTW) in the second stage. This aims to give a highly-accurate frame alignment, so that the DNN model training becomes more efficient.

4.1.2 Autoencoder Pretraining

Pretraining plays an important role in DNNs [23]. As the number of nodes in multiple layers ranges from hundreds to thousands, the resultant parameter space is enormous. Training a DNN with multiple hidden layers without pretraining is conventionally difficult. In the following, we propose a new pretraining technique in voice conversion by merely using the source spectral frames as both the input and output vectors and employing the L1 norm constraint on DNN weights. This is similar to training an autoencoder with sparsity constraint [20]. Autoencoder generally has much smaller number of nodes at the hidden layer in order to have dimension reduction and prevent the network from learning the identity function. However, as this autoencoder training acts as the pretraining for our spectral conversion DNN model, the hidden layers are not made to be narrow. Instead, we keep the same network architecture as of the DNN conversion network and use L1 norm constraint on DNN weights to employ a soft constraint on the sparsity of the activations of the hidden nodes.

This pretraining is motivated by the fact that both source and target spectral frames are aligned and in the same spoken content. The spectral shapes represented in the two features are roughly similar; the inter-difference may represent the speaker characteristics. Hence, by training an autoencoder of the source features, this provides a concise and effective initialization of the model parameters, which is believed to approximate the coarse spectral shape of the target feature.

At the beginning of this pretraining, all network weights are randomly initialized with uniform distribution and all biases are initialized at zero. The training criterion is the sum of squared error. After the pretraining error has converged, we remove the L1 norm constraint and use current network weights and biases as the initialization for the subsequent DNN model training on source and target features.

Note that this autoencoder pretraining for voice conversion is different from the layer-wise autoencoder pretraining in [41], where the autoencoder hidden representations given by the current layer are used as the input to the next layer. In our autoencoder pretraining for voice conversion, all parameters in the whole DNN are updated simultaneously. RBM pretraining is another popular pretraining method for DNN training. However, the optimization algorithm Contrastive Divergence [14] that

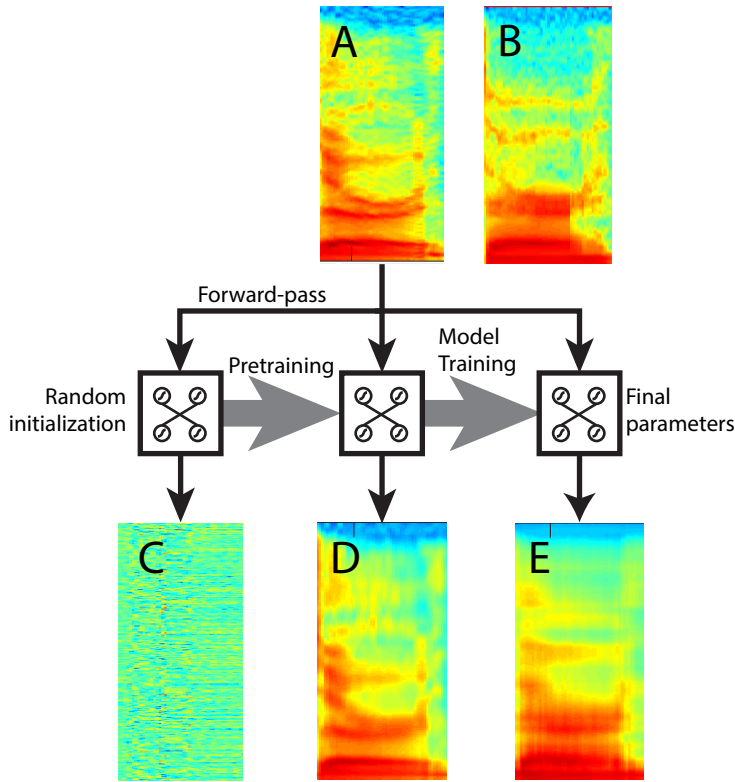


Fig. 3 Performing the forward-pass with the DNNs in different stages to demonstrate the effectiveness of autoencoder pretraining. A: spectrogram from source speech; B: spectrogram from target speech; C: output spectrogram from DNN with randomly initialized parameters; D: output spectrogram from DNN after autoencoder pretraining; E: output spectrogram from DNN after model training.

is used for RBM pretraining is unsupervised and not targeted on reconstructing spectral parameters. Moreover, comparing to discriminative layer-wise pretraining, autoencoder pretraining does not require early termination, which is usually required in discriminative layer-wise pretraining to prevent overfitting problem on each hidden layer.

In typical DNN training, there are three consecutive stages: initialization, pre-training and model training (fine-tuning). DNN-based voice conversion systems normally use only network after model training for conversion and evaluation. However, to demonstrate the effectiveness of the proposed pretraining, we would like to look at the output of the model in all the three stages, which are showed in Fig. 3. Fig. 3A and 3B are the source and target spectrograms respectively. In Fig. 3C, the output spectrogram from a DNN with randomly initialized parameters loses all the spectral information, showing that the DNN exhibits a very inaccurate estimation of the spectral transformation function. In Fig. 3D, after the autoencoder pretraining step, the output spectrogram get much closer to the source spectrogram, which indicates a much

more meaningful transformation represented by the DNN parameters. In Fig. 3E, after further training with target speech, the output spectrogram get closer to the target spectrogram in only a few number of epochs. In this example, we use 40 epochs for autoencoder pretraining and 20 epochs for model training.

4.2 F0 Trajectory Conversion

In our F0 conversion, we aim to model the transformation between the trajectories of source F0 and target F0. Since F0 is a prosodic feature that generally depends on long context information [25], we performed a segment-level conversion instead of frame-level scheme as in the above spectrum conversion and other typical DNN-based voice conversion [47].

The segments are defined by the voiced-unvoiced boundaries, where each segment is a continuous F0 measurement in voiced frames. The length of those segments are first normalized into a predefined number L before the segments are passed to the DNN model. The length normalization is done by interpolating the extracted F0 segment into a fixed length (This normalized length can be chosen empirically from the distribution of segment length observed in training data). Given the suprasegmental nature of F0 and the training amount of voice conversion, in particular the amount of voiced segments, is usually small, this length normalization is important that F0 trajectory is concisely captured and compact modeling is facilitated.

To model the F0 contour, rather than the F0 values itself, our feature vector for modeling $t'_{i,j}$ is formed by,

$$t'_{i,j} = \begin{cases} 0, & \text{if } i = 1 \\ t_{i,j} - t_{i-1,j}, & \text{if } i > 1. \end{cases} \quad (5)$$

where $t'_{i,j}$ is the i -th element of the j -th modeled feature vector. $t_{i,j}$ is the i -th element in the j -th length-normalized F0 segment of the target speaker. This is motivated by our previous findings on F0 modeling that relative pitch feature is effective for singing voice synthesis [21] and small amount of training data: Multiple segments could have different absolute F0 levels while their F0 trajectories are similar. Modeling the F0 contour allows those segments to share model parameters, which would alleviate the limitation of small amount of training data.

The resultant model output is an F0 trajectory, such that a reference mean level of the target F0 segment is necessary for the final F0 reconstruction. In this work, this target segmental mean level is calculated using the converted segment value from a frame-based F0 conversion system [47]. Other ways of calculating the segmental mean level are also possible. The reconstruction process from output segments is done in following three steps:

1. An F0 trajectory is reconstructed from output feature vector, assuming that the first value is 0.

$$\hat{t}_{i,j} = \begin{cases} 0, & \text{if } i = 1 \\ \hat{t}_{i-1,j} + \tilde{t}'_{i,j}, & \text{if } i > 1. \end{cases} \quad (6)$$

where $\tilde{t}'_{i,j}$ is the i -th element in the j -th output segment from the DNN model.

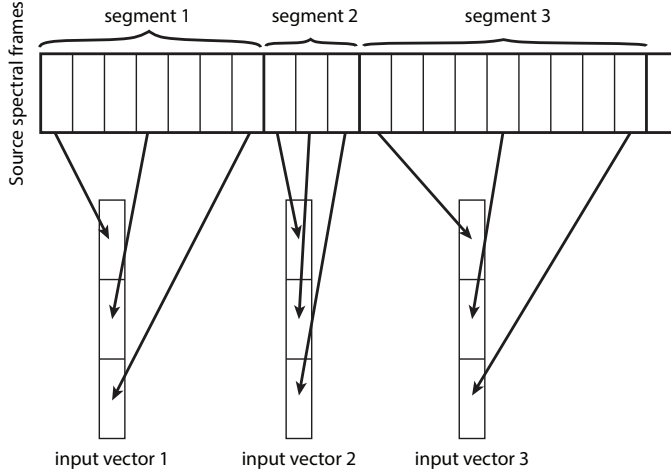


Fig. 4 Re-sample spectral segment to create input vector for duration modeling

2. Mean adjustment is applied to each reconstructed trajectory

$$\hat{t}_{i,j}^* = \hat{t}_{i,j} - \frac{\sum_{i=1}^L \hat{t}_{i,j}}{L} + \hat{\mu}_j \quad (7)$$

where $\hat{\mu}_j$ is the predicted segmental mean level of the j -th segment and $\hat{t}_{i,j}^*$ is i -th element of the j -th the mean-adjusted F0 segment.

3. The resultant segment is interpolated again into original length.

4.3 Intensity Trajectory and Phone Duration Conversion

For intensity, we perform a similar segmentation process as in F0 trajectory conversion, whereas intensity trajectories in voiced segments are extracted and then normalized in length. However, we choose to use intensity values directly in input and output feature vectors for DNN training in this work. Based on our informal experiments, similar performance is observed when using the intensity ratio or difference between consecutive frames as the input features. At conversion, converted spectrum will be scaled using the predicted intensity trajectory.

For duration conversion, we aim to model the duration ratio between the source and target segment. By segment here, we refer to a contiguous spectral block associated with a phone, a syllable, etc. The input feature is constructed by sampling spectral frames from this source segment. This is demonstrated in Fig. 4. In this example, each segment is constituted by three sampled frames. The number of sampled frames can be chosen empirically from the distribution of segment duration in the training data.

Without duration modification, all converted features has the same time alignment as the corresponding utterances from source speaker. Hence, we can employ the original segment timing of source speaker and the predicted phone duration ratio

from the DNN model to modify the segment length by performing linear interpolation on all converted features. In this work, we explore the performance of duration modeling with phone-based segment units.

5 Experiments

5.1 Experimental Setup

To evaluate the performance of the proposed works, we carried out several experiments on the CMU ARCTIC database [18]. In our experiments, we used four speaker pairs:

- BDL (male) to SLT (female)
- CLB (female) to RMS (male)
- CLB (female) to SLT(female)
- RMS (male) to BDL (male)

200 utterances in the dataset were used as training set and 60 other utterances were used as testing set. We used TANDEM-STRAIGHT [17] to calculate 512th-order log spectral envelope, voiced-unvoiced flag (VUV) and F0 with 5ms frame shift and 1024-point FFT. These speech features were then used to create different input and output vectors for our conversion models.

The detailed descriptions of each input, output features and network architectures are described below:

- **Spectrum conversion:** We used 512th-order log spectral envelope and its delta and delta-delta coefficients and one dimension of VUV from source speaker as input feature vector and 512th-order log spectral envelope from target speaker as output feature vector. The DNN had three hidden layers; each hidden layer had 3000 nodes.
- **F0 trajectory conversion:** We extracted F0 input and output segments as described in Section 4.2 with the normalized segment length L of 55. The DNN had two hidden layers; each hidden layer had 500 nodes.
- **Intensity trajectory conversion:** Input and output vectors extracted as described in Section 4.3 with the normalized segment length of 55. The DNN architecture is the same as in F0 trajectory conversion stated above.
- **Phone duration conversion:** For each phone in source utterances, an input vector was created by concatenating five spectrum frames from the phone, and the output vector was the ratio between the current source phone duration and the corresponding target phone duration. The phone boundaries can be determined by any phone recognizer. In our work, we use the boundaries from CMU Sphinx recognizer [19].

All the DNNs in this work employed hyperbolic tangent activation in the hidden nodes and there was no non-linear activation function in any output nodes. In all the system below, aperiodicity feature was not converted. We use the same aperiodicity feature from the source speaker to reconstruct the converted speech.

5.2 Spectrum Conversion

We evaluate the performance of DNN-based spectral conversion using the following systems:

- **JD-GMM**: The spectral representation was 25-th order MCEP, and the training and conversion was done as described in Section 2. We used 64 Gaussian components to build the system, which is empirically found in our preliminary experiments.
- **DNN-MCEP**: This system also used 25-th order MCEP as source and target feature. For DNN architecture and training parameters, we used the same settings as in [7]: two hidden layers with 50 nodes each layer, the learning rate was 0.01 and the momentum was 0.3.
- **DNN-SP256-DLP**: We built a system based on [46], in which the input vectors include log spectral frames with delta and delta-delta coefficients, log F0 with delta and delta-delta of the current frames and two context frames and one dimension of VUV. The DNN had two hidden layers with 1600 nodes each layer. We applied discriminative layer-wise pretraining (DLP) on this system.
- **DNN-SP512** (proposed): We used a DNN with the architecture described in Section 5.1. We did not perform any pretraining on this system.
- **DNN-SP512-DLP** (proposed): We used a DNN with the architecture described in Section 5.1 and applied discriminative layer-wise pretraining (DLP) on this system.
- **DNN-SP512-Autoencoder** (proposed): We used a DNN with the architecture described Section 5.1 and performed autoencoder pretraining as described in Section 4.1. The training data for the pretraining phase were 200 utterances from source speaker. This same set was also used in the model training stage.

5.2.1 Objective evaluation

We conducted objective evaluation to assess the proposed spectral conversion with autoencoder pretraining. Log spectral distortion (LSD) [48] was employed. The distortion of k -th order of log spectral pair is calculated as:

$$d(\mathbf{x}_k, \mathbf{y}_k) = \sum_{i=1}^M (\log(x_{k,i}) - \log(y_{k,i}))^2 \quad (8)$$

where M is the total number of frequency bin. A distortion ratio between converted-to-target distortion and the source-to-target distortion is defined as:

$$\text{LSD} = \frac{\sum_{k=1}^K d(\hat{\mathbf{x}}_k, \mathbf{y}_k)}{\sum_{k=1}^K d(\mathbf{x}_k, \mathbf{y}_k)} * 100\% \quad (9)$$

where \mathbf{x}_k , \mathbf{y}_k and $\hat{\mathbf{x}}_k$ denote the k -th frequency bin of the source, target and converted spectrum, respectively, and K denotes the number of frequency bins. A lower LSD indicates smaller distortion.

Table 1 presents the LSD for the baseline methods and our proposed methods. The average LSD over all evaluation pairs was reported. Comparing JD-GMM and

Table 1 Comparison of LSD ratio of different spectral conversion methods

| Methods | LSD (%) |
|------------------------------|--------------|
| JD-GMM | 87.93 |
| DNN-MCEP | 80.70 |
| DNN-SP256-DLP | 52.51 |
| DNN-SP512 | 54.73 |
| DNN-SP512-DLP | 52.59 |
| DNN-SP512-Autoencoder | 51.31 |

DNN-MCEP with other methods, we first observed that the systems working on high dimensional representations of spectrum (i.e. DNN-SP256-DLP, DNN-SP512, DNN-SP512-Autoencoder) yield much lower distortion than those working on low dimensional representations (at least 25% lower).

Comparing among DNN-SP256-DLP, DNN-SP512 and DNN-SP512-DLP, DNN-SP512 and DNN-SP512-DLP achieve a slightly higher distortion than DNN-SP256-DLP, that is 54.73% and 52.59% to 52.51%, which might indicate the difficulty in achieving a good minimum in training DNN with high dimensional features. However, when employing our proposed autoencoder pretraining, DNN-SP512-Autoencoder converges to a better solution with a lower LSD, and has a slightly lower distortion than DNN-SP256-DLP, that is 51.31% to 52.51%. This indicates that the proposed autoencoder pretraining enables a better initialization for spectral conversion.

Note that discriminative layer-wise pretraining was applied in DNN-SP256-DLP; while in our proposed DNN-SP512-Autoencoder, autoencoder pretraining was implemented. Comparing the LSD performance of these two pre-trained methods, the combination of autoencoder pretraining and DNN-SP512 (DNN-SP512-Autoencoder) yields a lower LSD with a doubled resolution in frequency.

5.2.2 Subjective evaluation

We conducted listening tests to access speech quality and speaker similarity of the converted results. 10 subjects participated in all listening tests. In this subjective evaluation, we compared DNN-SP512-Autoencoder with DNN-SP256-DLP, DNN-MCEP and JD-GMM. We did not include DNN-SP512 nor DNN-SP512-DLP due to the highly similar structure with DNN-SP512-Autoencoder.

We first performed AB preference tests to access speech quality. 20 pairs were randomly selected from 80 paired samples. In each pair, A and B were the samples from the proposed method and a baseline method, respectively, presented to listeners in a random order. Each listener was asked to listen to both samples and decide which sample was better in term of quality.

We then conducted XAB tests to access the speaker similarity. Similar to the AB test, 20 pairs were randomly selected from the 80 paired samples. In each pair, X was the reference target sample, and A and B were the converted samples of comparison methods listed in the first column of Table 2, presented to listeners in a random order. The listeners were asked to listen to the sample X first, then A and B, and then decide

Table 2 Results of average quality and similarity preference tests with 95% confidence intervals of different spectral conversion methods

| Methods | Preference score (%) | |
|------------------------------|----------------------|--------------------|
| | Quality test | Similarity test |
| JD-GMM | 35.8 (± 6.7) | 36.8 (± 8.3) |
| DNN-SP512-Autoencoder | 64.2 (± 6.7) | 63.2 (± 8.3) |
| DNN-MCEP | 42.5 (± 3.8) | 42.0 (± 6.6) |
| DNN-SP512-Autoencoder | 57.5 (± 3.8) | 58.0 (± 6.6) |
| DNN-SP256-DLP | 46.6 (± 3.8) | 44.0 (± 6.4) |
| DNN-SP512-Autoencoder | 53.3 (± 3.8) | 56.0 (± 6.4) |

which sample was closer to the reference target sample. For each pair of samples, X, A and B have the same language content.

The subjective evaluation results are presented in Table 2. Comparing to JD-GMM, our proposed method DNN-SP512-Autoencoder achieves much higher preference score in both quality and similarity test. Comparing to DNN-MCEP, DNN-SP512-Autoencoder also yields better results in both quality and similarity evaluations. Note that the spectral resolution in DNN-SP512-Autoencoder is much higher than JD-GMM and DNN-MCEP (In JD-GMM and DNN-MCEP, spectral envelope is represented by 25th-order MCEP). Spectral details are clearer in DNN-SP512-Autoencoder.

In the comparison between the state-of-the-art DNN-SP256-DLP and our proposed method DNN-SP512-Autoencoder, our proposed method achieves a slightly better subjective results in both quality and similarity. The results further confirms that DNN is capable of modeling high-resolution spectra and autoencoder pretraining can effectively improve the performance of DNN training for voice conversion.

5.3 Spectrum and F0 Conversion

We compared our proposed segment-based F0 conversion system with two baseline systems:

- **Mean-Var:** F0 is converted using the conventional global mean-variance transformation from source F0 [34] as follows,

$$f_o = \mu_t + \frac{\sigma_t}{\sigma_s} * (f_s - \mu_s). \quad (10)$$

where f_s and f_o are the source F0 and the output converted F0, respectively. μ_s , σ_s , μ_t , σ_t are the global mean and standard deviation statistics of the F0s from source and target speaker, respectively.

- **DNN-Frame:** F0 is modeled using a frame-based approach with DNN [47]. In this system, the input vector contains 256th-order log spectral envelope (static, delta and delta-delta), F0 (static values within a 7-element window, delta and delta-delta) and VUV. The DNN has two hidden layers with 1600 nodes each.

Table 3 Comparison of RMSE of different F0 conversion methods

| Methods | RMSE (Hz) |
|--------------------|-----------|
| Mean-Var | 22.06 |
| DNN-Frame | 17.90 |
| DNN-Segment | 17.80 |

Table 4 Results of average similarity preference tests with 95% confidence intervals of different F0 conversion methods

| Methods | Preference score (%) |
|--------------------|----------------------|
| | Similarity test |
| Mean-Var | 38.8 (± 2.1) |
| DNN-Segment | 61.2 (± 2.1) |
| DNN-Frame | 48.8 (± 4.1) |
| DNN-Segment | 51.2 (± 4.1) |

- **DNN-Segment** (proposed): The F0 segment is modeled as described in Section 4.2. The DNN architecture is described in Section 5.1.

In all F0 conversion systems, the result from DNN-SP512-Autoencoder was used as spectrum feature for speech reconstruction.

5.3.1 Objective evaluation

We employed root mean square error (RMSE) as the objective measurement to compare our proposed methods with the baseline systems. The average RMSE over all evaluation pairs was reported. A lower RMSE indicates smaller distortion.

Table 3 presents the RMSE for the baseline methods and our proposed method. Comparing to Mean-Var, DNN-Segment achieves lower RMSE, that is 17.80 to 22.06, and comparing to DNN-Frame, DNN-Segment achieves a slightly lower RMSE, that is 17.80 to 17.90, which confirms the effectiveness of the segment-based trajectory modeling method.

5.3.2 Subjective evaluation

We performed further evaluation with subjective listening test. We conducted an XAB similarity preference test in this evaluation. The setup is similarly to the XAB test in Section 5.2.2. We instructed the listeners to choose the sample whose prosody sound more similar to the reference target sample.

The subjective evaluation results are presented in Table 4. The similarity tests further confirm that our proposed method DNN-Segment provides better F0 modeling than the conventional Mean-Var method by a significantly better preference score, that is 61.2 to 38.8. Our proposed segment-based method DNN-Segment also has comparable preference score with state-of-the-art F0 modeling using DNNs.

A demonstration of the results from F0 conversion method is showed in Fig. 5.

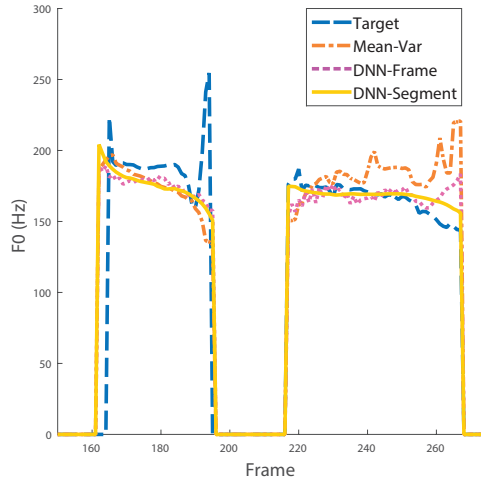


Fig. 5 Outputs of F0 conversion from three methods: Mean-Var, DNN-Frame and DNN-Segment

Table 5 Results of average similarity preference tests with 95% confidence intervals of the proposed intensity and duration conversion method

| Methods | Preference score (%) |
|------------------------------------|------------------------------------|
| | Similarity test |
| Without intensity and duration | 31.2 (± 4.1) |
| With intensity and duration | 68.8 (± 4.1) |

5.4 The Comprehensive Framework: Spectrum, F0 and Intensity and Duration Conversion

Finally, we combined our spectrum conversion and F0 conversion system with the proposed intensity and duration conversion described in Section 4.3. We compared the speaker similarity between the system with and without intensity and duration modification using listening tests. In this evaluation, we performed the evaluation on speaker pair CLB to SLT, where the intensity contour and time alignment between source and target are noticeably different.

The subjective evaluation results are presented in Table 5. The preference scores of systems with and without intensity and duration conversion are 68.8 and 31.2, respectively. This further suggests the effectiveness of our proposed system in converting prosodic features and the importance of having a comprehensive framework in voice conversion.

Some samples of this work are presented in the web link: <http://listeningtests.net/voiceconversion/hynq2015comprehensive>.

6 Conclusions

This paper showcases a comprehensive voice conversion framework, in which high-resolution spectra, F0, intensity and segment duration are all converted using DNNs. The objective and subjective evaluations shows the capability of the modeling of high-dimensional full spectra and prosodic segments. The proposed autoencoder pre-training for voice conversion is shown to effectively initialize the model parameters and leads to accurate spectral estimation. The resultant spectral conversion model with our proposed pretraining achieves comparable ratings in terms of similarity and quality as another state-of-the-art pretrained voice conversion system [46].

In conventional works of voice conversion, prosodic features are not commonly modeled. Knowing that human manipulates prosody in various levels, rather than in a local frame basis, we have introduced segment-level modeling and conversion for F0, intensity and duration, without increasing the amount of training data of typical voice conversion scenarios. Our experimental results have shown that the resultant converted speech is much similar to the target speaker. Throughout this work, a feasible conversion framework is built where multiple features are transformed acoustically and future investigations on ways of modeling and conversion of timbre and prosodic features remain thought-provoking. To better understand the performance of intensity and duration conversion, we plan to evaluate the perceptual effect of intensity and duration modification on voice conversion through more intensive psychoacoustic experiments.

Acknowledgements This research is supported by the National Research Foundation, Prime Ministers Office, Singapore under its IDM Futures Funding Initiative and administered by the Interactive and Digital Media Programme Office.

References

1. Adami, A.G., Mihaescu, R., Reynold, D.A., Godjirey, J.J.: Modeling prosodic dynamics for speaker recognition. In: Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 788–791 (2003)
2. Barlow, M., Wagner, M.: Prosody as a basis for determining speaker characteristics. In: Proc. The Australian International Conference on Speech Science and Technology, pp. 80–85 (1988)
3. Chen, L.H., Ling, Z.H., Dai, L.R.: Voice conversion using generative trained deep neural networks with multiple frame spectral envelopes. In: Proc. INTERSPEECH, September, pp. 2313–2317 (2014)
4. Chen, L.H., Ling, Z.H., Liu, L.J., Dai, L.R.: Voice conversion using deep neural networks with layer-wise generative training. *IEEE Transactions on Audio, Speech and Language Processing* **22**(12), 1859–1872 (2014)
5. Chen, L.H., Ling, Z.H., Song, Y., Dai, L.R.: Joint spectral distribution modeling using restricted Boltzmann machines for voice conversion (August), 3052–3056 (2013)
6. Dahan, D., Bernard, J.M.: Interspeaker variability in emphatic accent production in French. *Language and speech* **39**(4), 341–374 (1996)
7. Desai, S., Raghavendra, E., Yegnanarayana, B., Black, A., Prahallad, K.: Voice conversion using artificial neural networks. In: Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 3893–3896 (2009)
8. van Donzel, M.E., Koopmans-van Beinum, F.J.: Evaluation of prosodic characteristics in retold stories in Dutch by means of semantic scales. In: Proc. EUROSPEECH, pp. 211–214 (1997)
9. Erro, D., Moreno, A., Bonafonte, A.: Voice conversion based on weighted frequency warping. *IEEE Transactions on Audio, Speech, and Language Processing* **18**(5), 922–931 (2010)

10. Erro, D., Navas, E., Hernaez, I.: Parametric voice conversion based on bilinear frequency warping plus amplitude scaling. *IEEE Transactions on Audio, Speech and Language Processing* **21**(3), 556–566 (2013)
11. Helander, E., Silen, H., Virtanen, T., Gabbouj, M.: Voice conversion using dynamic kernel partial least squares regression. *IEEE Transactions on Audio, Speech, and Language Processing* **20**(3), 806–817 (2012)
12. Helander, E., Virtanen, T., Nurminen, J., Gabbouj, M.: Voice conversion using partial least squares regression. *IEEE Transactions on Audio, Speech and Language Processing* **18**(5), 912–921 (2010)
13. Helander, E.E., Nurminen, J.: A novel method for prosody prediction in voice conversion. In: *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 4, pp. 509–512 (2007)
14. Hinton, G.: Training products of experts by minimizing contrastive divergence. *Neural computation* **14**(8), 1771–1800 (2002)
15. Hwang, H.T., Tsao, Y., Wang, H.M., Wang, Y.R., Chen, S.H.: Incorporating global variance in the training phase of GMM-based voice conversion. In: *Proc. Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, pp. 1–6 (2013)
16. Kain, A., Macon, M.: Spectral voice conversion for text-to-speech synthesis. In: *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 1, pp. 285–288 vol.1 (1998)
17. Kawahara, H., Morise, M., Takahashi, T., Nisimura, R., Irino, T., Banno, H.: TANDEM-STRAIGHT: a temporally stable power spectral representation for periodic signals and applications to interference-free spectrum, F0, and aperiodicity estimation. In: *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3933–3936 (2008)
18. Kominek, J., Black, A.W.: CMU ARCTIC databases for speech synthesis. Tech. rep. (2003)
19. Lamere, P., Kwok, P., Walker, W., Gouvêa, E.B., Singh, R., Raj, B., Wolf, P.: Design of the CMU Sphinx-4 decoder. In: *Proc. EUROSPEECH*, pp. 1181–1184 (2003)
20. Le, Q.V., Coates, A., Prochnow, B., Ng, A.Y.: On optimization methods for deep learning. In: *Proc. The 28th International Conference on Machine Learning (ICML)*, pp. 265–272 (2011)
21. Lee, S.W., Ang, S.T., Dong, M., Li, H.: Generalized F0 modelling with absolute and relative pitch features for singing voice synthesis. In: *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 429–432 (2012)
22. Lee, S.W., Wu, Z., Dong, M., Tian, X., Li, H.: A comparative study of spectral transformation techniques for singing voice synthesis. In: *Proc. INTERSPEECH*, September, pp. 2499–2503 (2014)
23. Ling, Z.H., Kang, S.Y., Zen, H., Senior, A., Schuster, M., Qian, X.J., Meng, H., Deng, L.: Deep learning for acoustic modeling in parametric speech generation. *IEEE Signal Processing Magazine* (April), 35–52 (2015)
24. Liu, L.J., Chen, L.H., Ling, Z.H., Dai, L.R.: Spectral conversion using deep neural networks trained with multi-source speakers. In: *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4849–4853 (2015)
25. Meyer, G.A.: *The Semantics of Stress and Pitch in English*. The Faculty Association, Utah State University (1961)
26. Nakashika, T., Takashima, R.: Voice conversion in high-order eigen space using deep belief nets. In: *Proc. INTERSPEECH*, August, pp. 369–372 (2013)
27. Nakashika, T., Takiguchi, T., Ariki, Y.: High-order sequence modeling using speaker-dependent recurrent temporal restricted Boltzmann machines for voice conversion. In: *Proc. INTERSPEECH*, September, pp. 2278–2282 (2014)
28. Narendranath, M., Murthy, H.A., Rajendran, S., Yegnanarayana, B.: Transformation of formants for voice conversion using artificial neural networks. *Speech communication* **16**(2), 207–216 (1995)
29. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning representations by back-propagating errors. *Nature* **323**, 533–536 (1986)
30. Sanchez, G., Silen, H., Nurminen, J., Gabbouj, M.: Hierarchical modeling of F0 contours for voice conversion. In: *Proc. INTERSPEECH*, September, pp. 2318–2321 (2014)
31. Shriberg, E., Ferrer, L., Kajarekar, S., Venkataraman, A., Stolcke, A.: Modeling prosodic feature sequences for speaker recognition. *Speech Communication* **46**(3-4), 455–472 (2005)
32. Sorin, A., Shechtman, S., Pollet, V.: Coherent modification of pitch and energy for expressive prosody implantation. In: *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4914–4918 (2015)
33. Srikanth, R.: Duration modelling in voice conversion using artificial neural networks. In: *Proc. The Annual International Conference on Systems, Signals and Image Processing*, pp. 556–559 (2012)

34. Stylianou, Y., Cappé, O., Moulines, E.: Continuous probabilistic transform for voice conversion. *IEEE Transactions on Speech and Audio Processing* **6**(2), 131–142 (1998)
35. Sundermann, D., Ney, H., Hoge, H.: VTLN-based cross-language voice conversion. In: *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pp. 676–681 (2003)
36. Takashima, R., Takiguchi, T., Arik, Y.: Exemplar-based voice conversion in noisy environment. In: *Proc. Spoken Language Technology workshop (SLT)*, pp. 313–317 (2012)
37. Tian, X., Wu, Z., Lee, S.W., Chng, E.S.: Correlation-based frequency warping for voice conversion. In: *Proc. 9th International Symposium on Chinese Spoken Language Processing (ISCSLP)*, pp. 211–215 (2014)
38. Tian, X., Wu, Z., Lee, S.W., Hy, N.Q., Chng, E.S., Dong, M.: Sparse representation for frequency warping based voice conversion. In: *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 01, pp. 4235–4239 (2015)
39. Toda, T., Black, A.W., Tokuda, K.: Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory. *IEEE Transactions on Audio, Speech and Language Processing* **15**(8), 2222–2235 (2007)
40. Tokuda, K., Yoshimura, T., Masuko, T., Kobayashi, T., Kitamura, T.: Speech parameter generation algorithms for HMM-based speech synthesis. In: *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 3, pp. 1315–1318 (2000)
41. Vincent, P., Laroche, H., Lajoie, I., Bengio, Y., Manzagol, P.: Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research* **11**, 3371–3408 (2010)
42. Wu, C.H., Hsia, C.C., Liu, T.H., Wang, J.F.: Voice conversion using duration-embedded Bi-HMMs for expressive speech synthesis. *IEEE Transactions on Audio, Speech and Language Processing* **14**(4), 1109–1116 (2006)
43. Wu, Z., Chng, E.S., Li, H.: Conditional restricted Boltzmann machine for voice conversion. In: *Proc. IEEE China Summit & International Conference on Signal and Information Processing (ChinaSIP)*, pp. 104–108 (2013)
44. Wu, Z., Chng, E.S., Li, H.: Joint nonnegative matrix factorization for exemplar-based voice conversion. In: *Proc. INTERSPEECH*, September, pp. 2509–2513 (2014)
45. Wu, Z., Virtanen, T., Chng, E.S., Li, H.: Exemplar-based sparse representation with residual compensation for voice conversion. *IEEE Transactions on Audio, Speech and Language Processing* **22**(10), 1506–1521 (2014)
46. Xie, F.L., Qian, Y., Fan, Y., Soong, F.K., Li, H.: Sequence error (SE) minimization training of neural network for voice conversion. In: *Proc. INTERSPEECH*, September, pp. 2283–2287 (2014)
47. Xie, F.L., Qian, Y., Soong, F.K., Li, H.: Pitch transformation in neural network based voice conversion. In: *Proc. The 9th International Symposium on Chinese Spoken Language Processing (ISCSLP)*, pp. 197–200. *IEEE* (2014)
48. Ye, H., Young, S.: High quality voice morphing. In: *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 1, pp. 1–9–12 (2004)
49. Yoshimura, T., Tokuda, K., Masuko, T., Kobayashi, T., Kitamura, T.: Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis. In: *Proc. EUROSPEECH*, pp. 2347–2350 (1999)
50. Yu, D., Deng, L.: Deep learning and its applications to signal and information processing. *IEEE Signal Processing Magazine* pp. 145–154 (2011)
51. Yu, D., Deng, L.: *Automatic Speech Recognition - A Deep Learning Approach*. Springer-Verlag London (2015)
52. Yutani, K., Uto, Y., Nankaku, Y., Toda, T., Tokuda, K.: Simultaneous conversion of duration and spectrum based on statistical models including time-sequence matching. In: *Proc. INTERSPEECH*, 3, pp. 1072–1075 (2008)